VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

MRI IMAGE SEGMANTATION BASED ON EDGE DETECTION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

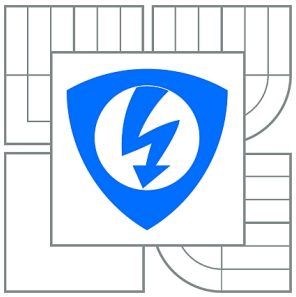
AUTOR PRÁCE
AUTHOR

MICHAIL SIMICHANIDIS

BRNO 2012

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF BIOMEDICAL ENGINEERING

MRI IMAGE SEGMANTATION BASED ON EDGE DETECTION

SEGMENTACE MRI DAT S VYUŽITÍM DETEKCE HRAN

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

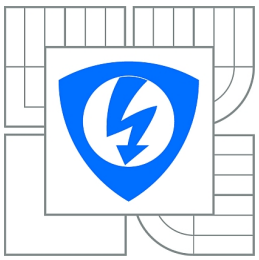
AUTOR PRÁCE — MICHAIL SIMICHANIDIS

AUTHOR

VEDOUCÍ PRÁCE — Ing. VRATISLAV HARABIŠ

SUPERVISOR

BRNO 2012



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav biomedicínského inženýrství

Bakalářská práce

bakalářský studijní obor
Biomedicínská technika a bioinformatika

Student:	Michail Simichanidis	***ID:***	126759
Ročník:	3	***Akademický rok:***	2011/2012

NÁZEV TÉMATU:

Segmentace MRI dat s využitím detekce hran

POKYNY PRO VYPRACOVÁNÍ:

1) Proveďte literární rešerši v oblasti segmentace obrazových 3D dat z magnetické rezonance. Zaměřte se zejména na metody využívající detekci hran. 2) Navrhněte jednoduchý prohlížeč 3D MRI dat a zásuvný modul pro segmentaci dat. 3) Ve vhodném prostředí implementujte prohlížeč 3D MRI dat a navrhněte vhodnou metodu pro segmentaci 3D MRI dat využívající detekce hran v obraze. 4) Navrženou metodu implementujte jako modul prohlížeče a proveďte její otestování na dostatečném počtu vzorků. 5) Vytvořte uživatelskou příručku k programu. 6) Proveďte diskuzi dosažených výsledků a vyhodnoťte funkčnost implementace.

DOPORUČENÁ LITERATURA:

[1] JAN, J. Medical Image Processing, Reconstruction and Restoration - Concepts and Methods. Signal Processing and Comm. Signal Processing and Comm. Boca Raton, FL, USA: CRC Press, Taylor and Francis Group, 2006. 760 s. ISBN: 0-8247-5849- 8

[2]TANG, H., WU, E. X., MA, Q. Y., GALLAGHER, D., PERERA, G. M., ZHUANG, T. MRI brain image segmentation by multi-resolution edge detection and region selection, Computerized Medical Imaging and Graphics, Volume 24, Issue 6, November 2000, Pages 349-357, ISSN 0895-6111, DOI: 10.1016/S0895-6111(00)00037-9

Termín zadání:	6.2.2012	***Termín odevzdání:***	25.5.2012

Vedoucí práce: Ing. Vratislav Harabiš
Konzultanti bakalářské práce:

prof. Ing. Ivo Provazník, Ph.D.
Předseda oborové rady

ABSTRACT

Cílem této práce je představit základní segmentační techniky používáné v oblasti medicínského zpracování obrazových dat a pomocí 3D prohlížeče schopného zobrazit 3D obrazy implementovat segmentační modul založený na hranové detekci a vyhodnotit výsledky. Navrhovaný prohlížeč je sestavený v prostředi Matlab GUI a je schopen načíst objem 3D snímků představující lidskou hlavu. Navrhovaný segmentační modul je založen na použití hranových detektorů, zejména Cannyho detektoru.

KEYWORDS

DICOM, MRI, Segentation, Edge, Sobel, Prewitt, Canny

ABSTRAKT

The aim of this thesis is to present the basic segmentation techniques uses in the field of medical image processing and by using a 3D viewer able to visualize 3D images, implement a segmentation module based on edges detection and evaluate the results. The proposed viewer is a 3D viewer build using matlab GUI and is able to load a volume of images representing the human head. The proposed segmentation module is based on the use of edge detectors particularly the Canny algorithm.

KLÍČOVÁ SLOVA

Překlad klíčových slov v angličtině nebo češtině

SIMICHANIDIS, Michail *SEGMENTACE MRI DAT S VYUŽITÍM DETEKCE HRAN*: bachelor's thesis. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, Ústav biomedicínského inženýrství, 2012. 42 p. Supervised by Ing. Vratislav Harabiš

DECLARATION

I declare that I have elaborated my bachelor's thesis on the theme of "SEGMENTACE MRI DAT S VYUŽITÍM DETEKCE HRAN" independently, under the supervision of the bachelor's thesis supervisor and with the use of technical literature and other sources of information which are all quoted in the thesis and detailed in the list of literature at the end of the thesis.

As the author of the bachelor's thesis I furthermore declare that, concerning the creation of this bachelor's thesis, I have not infringed any copyright. In particular, I have not unlawfully encroached on anyone's personal copyright and I am fully aware of the consequences in the case of breaking Regulation § 11 and the following of the Copyright Act No 121/2000 Vol., including the possible consequences of criminal law resulted from Regulation § 152 of Criminal Act No 140/1961 Vol.

Brno ....

(author's signature)

ACKNOWLEDGEMENT

I would like to thank my supervisor on this thesis Ing. Vratislav Harabiš for his professional guidance and consultation. I would also like thank the diagnostic center Asklipios Medica Giannitsa and his head technologist Paulos Iordanidis for inspiring this thesis.

Brno ....

(author's signature)

CONTENTS

LIST OF FIGURES

INTODUCTION

Today, it is possible to come across with many different medical imaging systems, but in my work, I will deal only with magnetic resonance imaging. Magnetic resonance imaging is currently one of the most used and most accurate method of displaying structures of the human body. It provides us with detailed anatomical structures without influencing cell structures using radiation.

Input data are DICOM files. There are many different approaches to segmentation, so I describe some basic techniques that we can encounter in biomedicine. They can be classified according to the features and the technique used. The features include texture, gray values, and gradient magnitudes. Segmentation techniques can be classified as either non-contextual or contextual. Non-contextual techniques do not take in account the relationships that exist between features in an image. So pixels are grouped together on the basis of a global attribute, like gray level. For example intensity-based thresholding, where each pixel is assigned to a particular region based on its gray value, is a non contextual technique [2], [9].

Contextual techniques make use of the relationships between image features. Thus, a contextual technique might group together pixels that have more or less the same gradient values and are close to one another. Such techniques may include: region-based techniques, such as region growing and watershed .And boundary-based techniques, where edge-based methods are used to outline the boundaries between regions [2], [9].

1 MAGNETIC RESONANCE IMAGING

The principles of MRI rely on the spinning motion of specific nuclei present in biological tissues. These are known as MR active nuclei and are characterized by their tendency to align their axis of rotation to an external magnetic field. Nuclei that have a net charge and are spinning acquire a magnetic moment which we can think of as vector, the magnetization vector and are able to align with a external magnetic field Bo. This occurs only if the mass number of the atomic nuclei is odd. The interaction of the magnetization vector with the external magnetic field Bo is the basis of MRI [4].

The hydrogen nucleus is the MR active nucleus used in clinical MRI. Hydrogen is used because it is abundant in the human body, and because its solitary proton gives it a relatively large magnetic moment. Other examples of MR active nuclei with odd mass number are carbon, nitrogen, oxygen etc.

Hydrogen nuclei possesses only two energy states, we can call them low energy state and high energy state. In the absence of an applied magnetic field the magnetic moments or else magnetization vectors are randomly oriented. But when placed in a external magnetic field they tend to align with it. The low energy nuclei align their magnetic moments parallel to the external field and the high energy nuclei align in the anti-parallel direction. So when a patient is placed inside a magnetic field in our case inside the bore of the magnet, the hydrogen nuclei within the patient align parallel and anti-parallel to Bo. A small excess of hydrogen nuclei align up parallel to Bo and constitute the net magnetization vector of the patient [6].

The net magnetization vector does not simply align with the Bo but wobbles around it creating a movement called precession. and the speed at which it wobbles around Bo is called precessional frequency. The precessional frequency is often called the Larmor frequency because it is determined by the Larmor equation [6]:

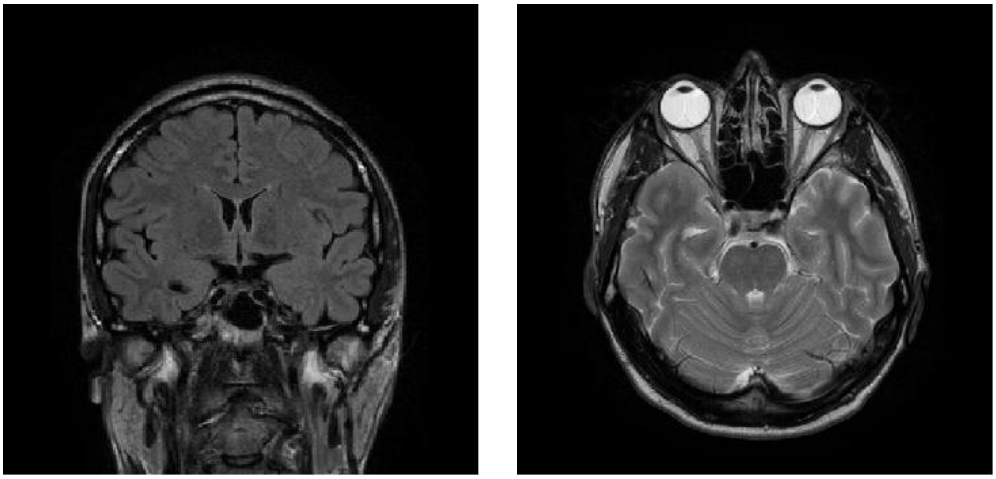
$$\omega = Bo * \gamma \tag{1.1}$$

where Bo is the field strength of the magnet, γ is the gyromagnetic ratio.

For resonance to occur we need an RF pulse of energy at exactly the Larmor frequency of the hydrogen. This absorption of energy causes the low energy hydrogen nuclei to gain energy via resonance and become high energy nuclei. Resonance has two immediate effects. The first result of resonance is that the net magnetization vector moves out of alignment away from Bo. The second result is that moves the magnetic moments of the hydrogen nuclei into phase. Signal is finally produced when coherent magnetization cuts across a coil [4].

2 STANDARD DICOM

Standard DICOM (Digital Imaging and Communications in Medicine) was developed from the NationalElectrical Manufacturers Association NEMA for the distribtion and visualization of medical data aquired from all kinds of medical imaging systems such as RTG, CT, MRI.



(a) Coronal plane (b) Axial plane

Fig. 2.1: DICOM

In short DICOM is an all-encompassing data transfer, storage, and display protocol built and designed to cover all functional aspects of digital medical imaging. It provides all the necessary tools for accurate representation and processing of medical imaging data such as full support for numerus image acquisition parameters. Other than just storing images DICOM also records a multitude of other image related parameters such as slice thickness, image exposure and patient 3D position. DICOM files and messages use standardized attributes to convey various medical data from patient name to image color depth, to current patient diagnosis. DICOM also provide excellent image quality and clarity in describing digital imaging devices and their functionality. Working with medical devices through their DICOM interfaces becomes a very straightforward process, leaving little room for errors [1].

The data used to demonstrate the various segmentation methods in the various chapters were provided by the department of radiology in Fakultni Nemocnice Brno, Czech Republic. The data that were provided are in DICOM format representing millimeter slices of a head in coronal an axial plane. The resolution for the axial slices is 512x512 and 320x320 and for the coronal slices is 256x256. The slides were taken on 29.06.2011 using a MRI Philips Archieva 1.5T.

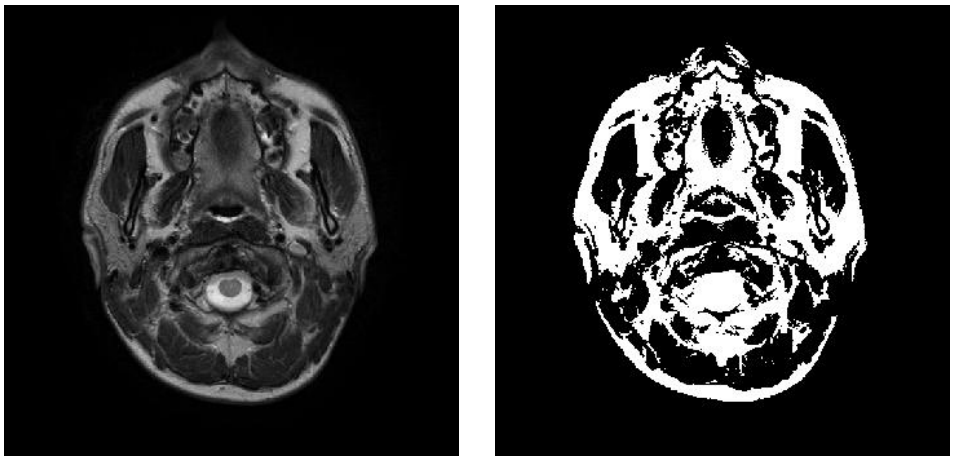
3 SEGMENTATION METHODS

3.1 Intensity-based thresholding

Thresholding as a segmentation method exploits a global attribute such as a threshold value so that pixels having values greater than the threshold are assigned to one region while those that fall below the threshold are assigned to another region. So it is basically intensity thresholding where each pixel is compared to the threshold creating thus two different regions. Pixels that have value higher than the threshold value are considered to be foreground and then pixel value is set to 1 while the pixels that have value smaller than the threshold value are considered to be background and their pixel value is set to zero. So thresholding creates a binary image b (x, y) from an intensity image (x, y) according to the criterion [3], [9]:

$$b(x,y) = \begin{cases} 1, & if(x,y) > T \\ 0, & otherwise \end{cases} \tag{3.1}$$

Where T is threshold



(a) original image (b) segmented image

Fig. 3.1: Thresholding

As shown in figure (3.1) in some cases using thresholding as means of segmentation can be satisfactory. The image shows the result of intensity thresholding where the thresholding value was chosen manually by trial and error.

But the technique is rather limiting. It is sensitive to accidental and uncontrolled variations in the illumination field and in the end it is only really applicable to those simple cases in which the entire image is divisible into a foreground of objects of similar intensity and a background of distinct intensity to the objects [3].

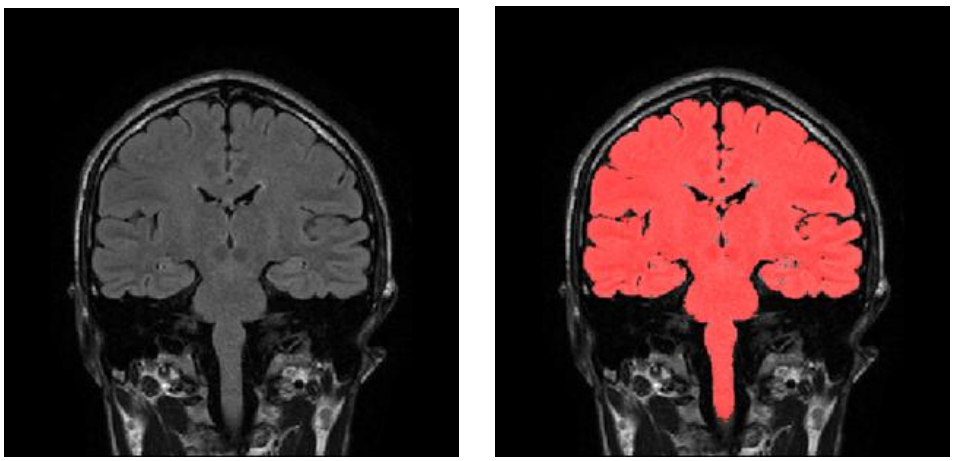
3.2 Region-based techniques

These are methods that seek to divide an image into as many homogeneous regions as possible. The regions are clusters of merged pixels of the same or similar properties, for example, same intensity or brightness level. Now days, there are many methods based on regions. These methods are very useful and effective particularly in image processing, in which there is a large amount of noise. Among the most important, we can include region growing, region merging, region splitting and [2], [9].

3.2.1 Region growing

This method produces the segmented image regions based on the similarity of pixels in the region. This similarity is one of the predefined criteria, unlike methods based on edge detection that look for differences in the image pixels.

The end result of this segmentation method are regions which are formed by clustering individual pixels, having some similarities between them. Unfortunetelly the similarities between pixels as the main criterion for segmentation would not be in itself too effective. It is also necessary to define several other criteria that would take into account the spatial relationships between neighboring pixels for example, their proximity [5], [9].



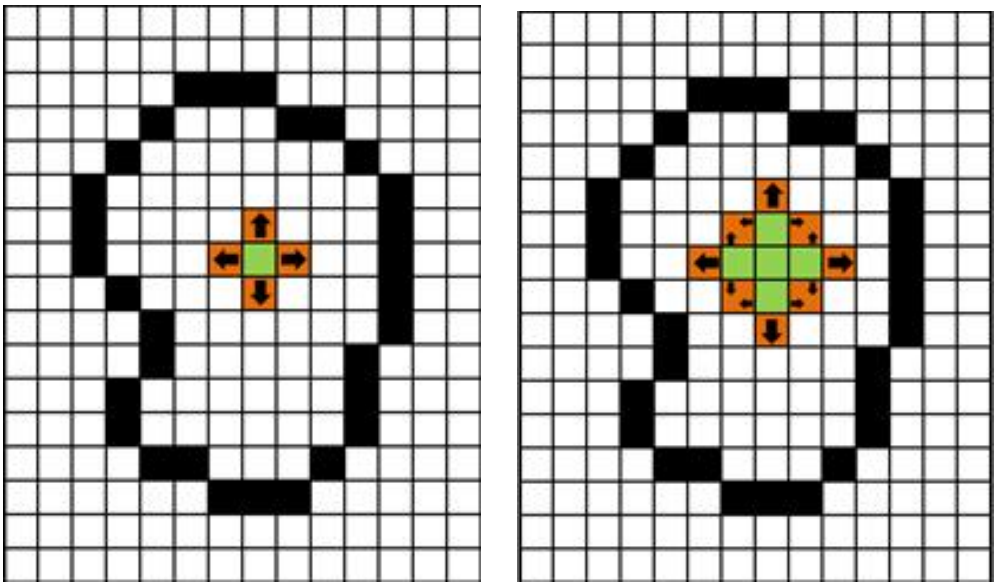
(a) Original image (b) segmented image

Fig. 3.2: Region growing

Normally when using segmentation methods such as region growing we create the initial regions using the so called ”seeds points”. The seeds are randomly selected pixels throughout the image. These pixels form the starting points of the region itself and start growing from these exact points,figure (3.3). Connecting neighboring pixels

to the created seed points creates a growing region until the similarity criterion is met. Examples of similarity criteria can be [2]:

- The absolute difference of intensities between pixel and seed pixel must be examined in the specified range.
- The absolute difference between the intensities examined and the average pixel intensity in the growing region must lie in the specified range.
- All pixels fall within a range of homogeneity.



(a) Region growing from seed point

(b) Gradual growth of the surrounding region

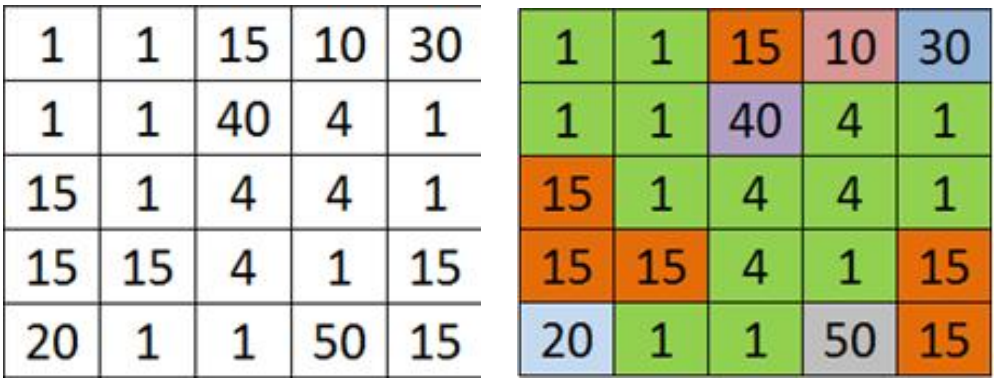
Fig. 3.3: Seed points

3.2.2 Region merging

By using region merging we are trying to eliminate false edges and false regions in an image so that the joining regions have common characteristics belonging to a particular object. So we join together small homogeneous areas in order to create a larger area, which will represent our region of interest. For region merging to work we need to meet several criteria of homogeneity [3]:

$$H(Ri) = TRUE\ pro\ i = 1, 2, \ldots I \tag{3.2}$$

Where I is the number of regions in the picture, Ri the defined region, H(Ri) is a two valued variable expressing the criterion of homogeneity.



(a) image divided into small areas (b) completion of the merged areas

Fig. 3.4: Region merging

As shown in figure(3.4), the image is first divided into very small regions (pixels) and then we compare them with each other so that if the absolute difference between two adjacent areas is less than or equal to 3 the pixel is compatible, and assigned a specified color.

3.3 Boundary-based techniques

Edge detection is considered to be one of the most important aspects of image processing. If we are able to find the boundry of an obejct by locating all its edges, then we can say that we have effectively segmented. Edge detection seems to be a rather easy process concedering that edges are simply regions of intensity transition between one object and another. Howerver it remains despite its simplicity an active field of research. Most edge detectors are based on the use of derevative filters. Derivative filters are used to detect discontinuities in an image. Derivative filters are designed to return significant values at points of discontinuity in the image and to give no response in perfectly smooth regions of the image, in other words they detect edges [9].

Edge detection makes use of differential operators for detecting changes in the gradients of the grey or color levels in the image. Edge detection is divided into two categories: first-order edge detection and second-order edge detection. First order edge detection is based on the use of first-order image derivatives, whereas second order edge detection is based on the use of second-order image derivatives. But before using the derivative filters we have to remember two things [3]:

1. Differentiation is a linear operation and a discrete approximation of a derivative filter can thus be implemented by the kernel. From the discrete approximations we must therefore, devise appropriate filter kernels to represent each

of the derivative operators.

2. A very important condition we must impose on such a filter kernel is that its response be zero in completely smooth regions. This condition can be enforced by ensuring that the weights in the kernel mask sum to zero.

3.3.1 First order Edge Detection

Methods based on the first derivative, are based on the idea that in places where the edge occurs, there is the greatest change in intensity, while in homogeneous areas the change and thus the first derivative is zero. First derivative methods are also called gradient methods, where we are looking for the maximum and minimum in the first derivative of the image function [7].

In other words, if we look at a image as a function, witch for each pixel returns an intensity value at a given point , we regard an edge the place of the image where there is a change of intensity. If we take the gradient of this signal we get the first derivative of the image function, figure (3.5).

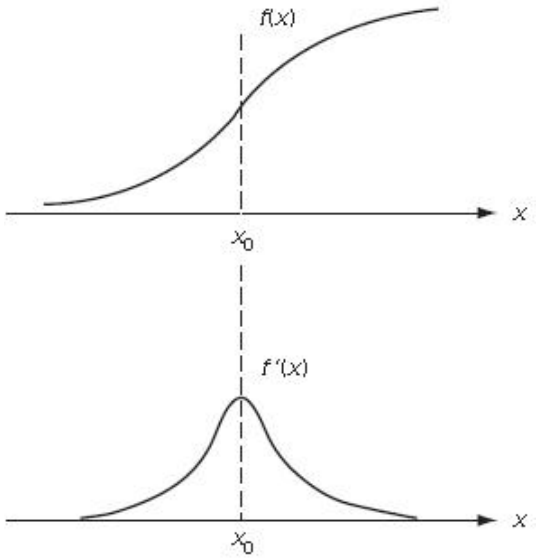


Fig. 3.5: First derivative of the image function[2]

The derivative shows a maximum located at the center of the edge in the original signal. This method of locating an edge is characteristic of the gradient filter family of edge detection filters. A pixel location is declared an edge location if the value of the gradient exceeds some threshold. As we have already mentioned, edges will have higher pixel intensity values than those surrounding it. So once a threshold is set, you can compare the gradient value to the threshold value and detect an edge whenever the threshold is exceeded [7].

Robert's cross operator

Robert's cross operator is a differential operator witch performs a simple 2D spatial gradient measurement on an image. In other words it highlights regions corresponding to edges. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation. These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient. So pixel values in the output image represent the estimated absolute magnitude of the spatial gradient of the input image. The operator consists of two 2x2 convolution kernels. One kernel is simply the other rotated by 90°. The Roberts cross operator is fast to compute, due to the minimal size of the kernels but it is very sensitive to noise [10].

$$\mathbf{G_x} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \mathbf{G_y} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \tag{3.3}$$

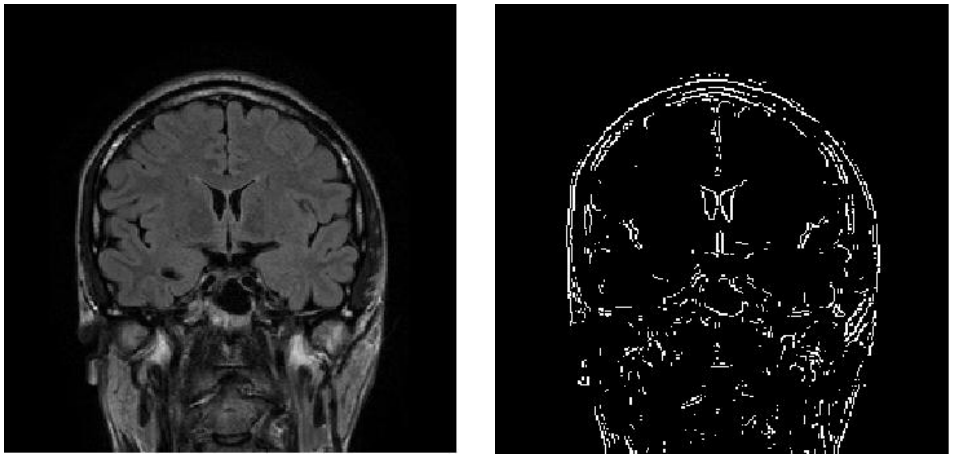
Sobel and Prewitt operators

In more complex images, operators such as Sobel and Prewitt operator will have a better result in finding edges in an input image, primary because of use of the more complex kernels . The Sobel operator performs a 2D spatial gradient measurement on an image and it is often used to calculate the absolute gradient magnitude at each point in an input image. The operator consists of a pair of 3x3 convolution kernels. These kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations [10].

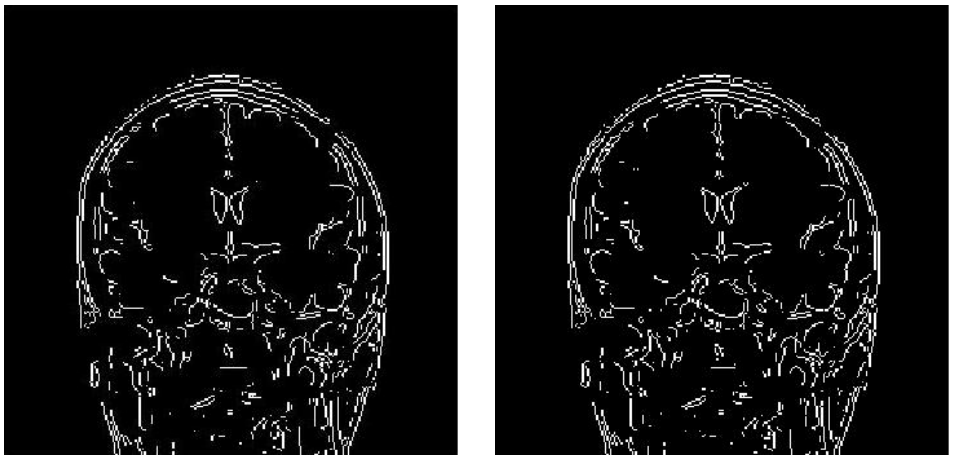
$$\mathbf{G_x} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \mathbf{G_y} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{3.4}$$

Prewitt operator is very similar to Sobel, but uses different kernels [10]:

$$\mathbf{G_x} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \mathbf{G_y} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \tag{3.5}$$



(a) original image (b) Roberts cross operator



(c) Prewitt (d) Sobel

Fig. 3.6: Edge-based segmentation

3.3.2 Second-Order Edge Detection

Although the application of a derivative operator is a vital step in image segmentation there is actually considerably more to robust edge detection than the simple application of a derivative kernel.A much more common means of image enhancement is through the use of a second-order derivative operators.

As shown in figure (3.7) , in places where the first derivative is maximum, that is to say the places where we have the biggest change in intensity, the second derivative passes through zero. So unlike the first derivative methods where we are looking for the maximum in the first derivative of the image we detect places where the second derivative passes through zero.

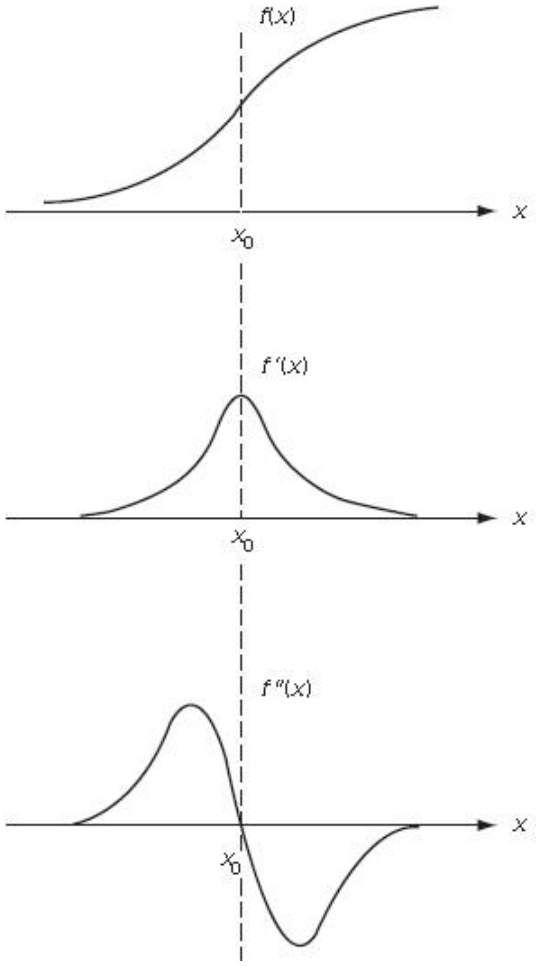


Fig. 3.7: Second derivative of the image function [2]

The Laplacian opertor

A very popular second-order derivative operator is the Laplacian.

The Laplacian operator is in essence a measure of the second spatial derivative of an image. The Laplacian of an image highlights regions of rapid intensity change and is often used for edge detection. The Laplacian is often applied to an image that has first been smoothed with something approximating a Gaussian Smoothing filter in order to reduce its sensitivity to noise. The operator normally takes a single gray level image as input and produces another gray level image as output [7].

The second-order derivative property that allows the Laplacian to produce a fine edge response corresponding to a change in gradient, rather than the less isolated response of the first-order edge filters, makes it suitable as the first stage of digital edge enhancement.

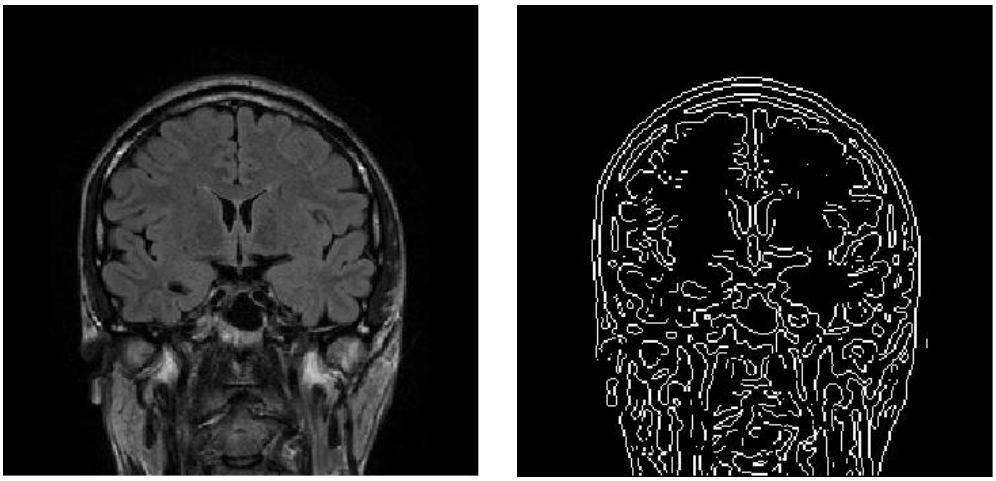
Because these kernels are approximating a second derivative measurement on the

image, they are very sensitive to noise. To counter this, the image is often Gaussian Smoothed before applying the Laplacian filter. This pre-processing step reduces the high frequency noise components prior to the differentiation step.

Since the convolution operation is associative, we can convolve the Gaussian smoothing filter with the Laplacian filter first of all, and then convolve this filter with the image to achieve the required result. Since both the Gaussian and the Laplacian kernels are usually much smaller than the image, this method usually requires far fewer arithmetic operations [7].

Laplacian of Gaussian

To counter this high noise sensitivity of the Laplacian filter, the standard Laplacian kernel is commonly combined with the Gaussian kernel to produce a robust filtering method. These two kernels could be applied sequentially to the image as two separate convolution operations. First smoothing with the Gaussian kernel and then with the Laplacian. However, as convolution is associative we can combine the kernels by convolving the Gaussian smoothing operator with the Laplacian operator to produce a single kernel: the Laplacian of Gaussian (LoG) filter. This single kernel is then applied to the image in a single pass. This offers a significant computational saving by approximately halving the calculations required. The response of the filter will be zero in areas of uniform image intensity, whilst it will be nonzero in an area of transition. At a given edge, the operator will return a positive response on the darker side and negative on the lighter side [7].



(a) original image (b) segmented image

Fig. 3.8: Laplacian of Gaussian

3.3.3 Canny

The Canny operators generally acknowledged as the best edge detection method developer so far. It has been used in many areas of digital imaging processing because of its simple algorithm which makes the whole process to be efficiently executed. It was designed from the beginning to be an optimal edge detector based on three strict criteria [3].

1. Low error rate: Edges occurring in mages should not be missed and there should be no response where edges do not exist.
2. The criterion of positioning accuracy: Report edge location at correct position. In other words the distance between the edge pixels as found by the detector and the actual edge has to be at a minimum
3. The criterion of single edge response: Reduce the number of responses to a single edge response

Based on those criteria, the canny edge detector smooths the input image by convolution with the Gaussian mask. Here it is good to point out that the larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise (i.e. noise reduction) is achieved. However, the localization error also increases slightly as the Gaussian width increases.

Following the smoothing of the image, it finds the strength and direction of the edges using for example, the Prewitt or Sobel operator. These operators highlight ridges with high spatial derivatives. This is achieved by taking the gradient of the image in the horizontal and vertical directions and then adding the magnitude of these components as a measure of edge strength. The direction of the edge is easily calculated using the gradient in the x and y directions. The algorithm then tracks along the top of these ridges and sets to zero (i.e. suppresses) all pixels that are not actually on the ridge top so as to give a thin line in the output, a process known as non-maximal suppression [7].

Finaly hysterisis is used as a method of preventing the edge contour breaking due to fluctuations around a single theshold. Instead, two thresholds, T1 and T2 with T1 $<$ T2, are used [7]. In order to implement the canny edge detector algorithm, a series of steps must be excuted.

The first step is to filter out any noise in the originál image using a Gaussian filter. Once a suitable mask has been calculated , the Gaussian smoothing can be performed using standart connvolution methods. In other words a mask , essentialy a small matrix of k x k elements is rotated through 180° and placed on top of the input image , starting at the top left position of the image . The mask elements are summed and suitably normilized to form a weighted response witch is then te value of the output pixel at the position corresponding to the center of the mask. The mask

is then moved one position to the right,the sum of products are re-calculated and normilized to give the next pixel value in the output image . This proces sis repeated as the mask is moved along the input image in a raster scan . After smoothing the image and minimizing the noise, the next step is to find the edge strength by taking the gradient of the image. This is done by using a first derivative operator like the Sobel operator. The operator consists of a pair of 3x3 convolution kernels:

These kernels are designed to respond maximally to edges horizontally and vertically relative to the pixel grid, one kernel for each of the orientations. The kernels in order to produce separate measurements of the gradient component in each orientation can be applied separately, or they can be combined together to find the absolute magnitude of the gradient at each point. The gradient magnitude is given by[7]:

$$\mid G \mid = \sqrt{Gx^2 + Gy^2}, \tag{3.6}$$

After computing the absolute gradient magnitude at each point we have to find the direction of the edge. The direction is computed with the help of the gradients Gx, Gy [7]:

$$\theta = \arctan\left(\frac{Gy}{Gx}\right) \tag{3.7}$$

The next step is to digitize the edge direction so that it can be traced in an image. In other words we have to approximate the known edge direction to one that can be traced in digital image. So if the pixels of a 5x5 image are aligned as follows: It

Fig. 3.9: Pixel alignment

can be seen by looking at pixel "a ", there are four possible directions describing the surrounding pixels - 135 degrees witch is along the negative diagonal, 45 degrees witch is along the positive diagonal, 0 degrees witch is in the horizontal direction and 90 degrees witch is in the vertical direction. So now the edge orientation has to be resolved into one of these four directions depending on which direction it is closest to so for example if the orientation angle is found to be 20 degrees, we will make it zero degrees. We can think this as taking a semicircle and dividing it into 5 regions.

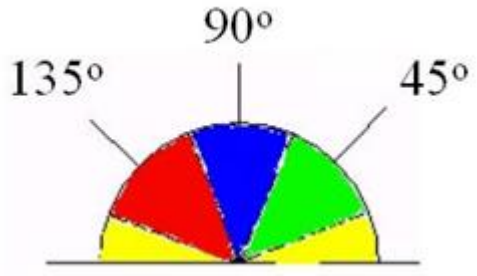


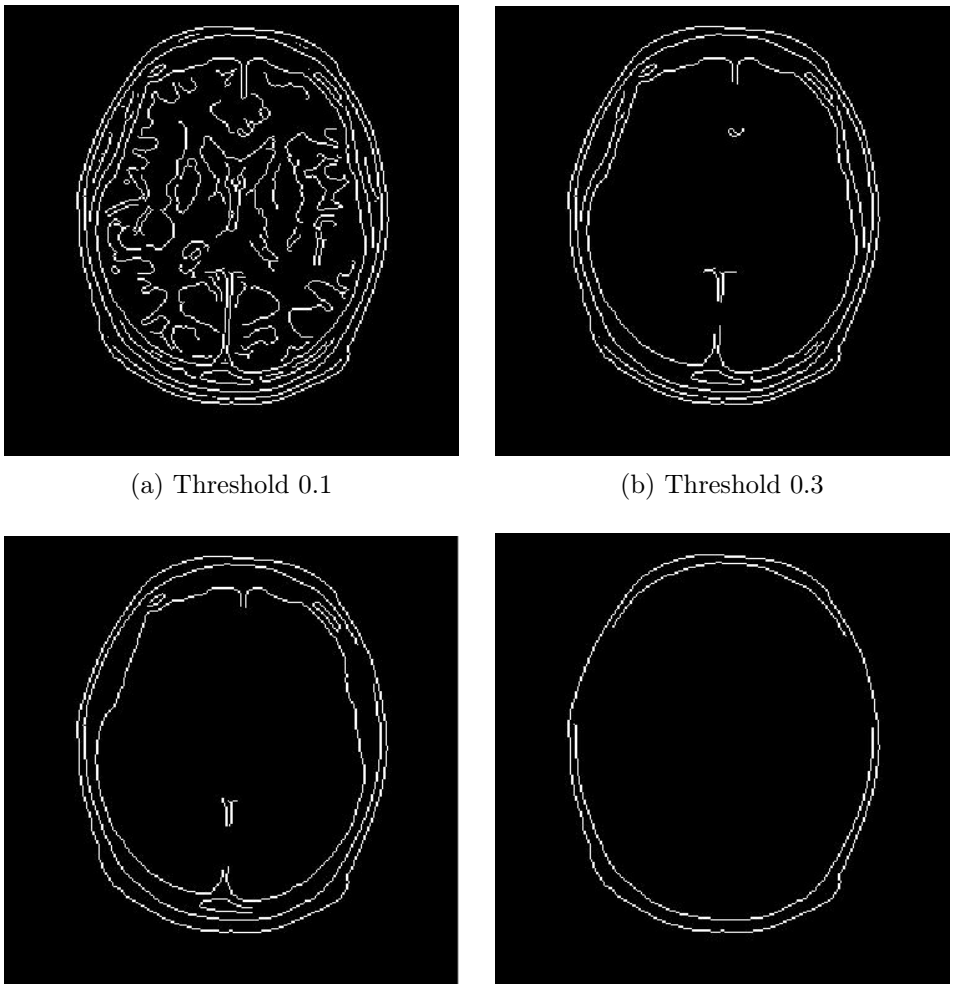
Fig. 3.10: Pixel digitization [7]

So, any edge direction falling within the yellow rangewitch ranges 0 to 22.5 and 157.5 to 180 degrees is set to 0 degrees. Any edge direction falling in the green range witch corresponds to 22.5 to 67.5 degrees is set to 45 degrees. Any edge direction falling in the blue range 67.5 to 112.5 degrees is set to 90 degrees. And any edge direction falling within the red range 112.5 to 157.5 degrees is set to 135 degrees.

After the edge directions are known, no maximum suppression is applied. This works by tracing along the edge in the edge direction and suppressing any pixel value, that is to say it sets any pixel value to zero that is not considered to be an edge. This will give a thin line in the output image. Thsi is simply done by comparing the edge strength of the current pixel with the edge strength of the pixel in the positive and negative gradient direction. So for example, if the gradient directioin is north , we compare with the pixels to the north and south [7].

After the first five steps have been completed, the final step is to track along the remaining pixels that have not been suppressed and threshold the image to identify the edge pixels. To accomplish that we use two distinct thresholds, a higher value T2 and a lower value T1. The fate of each pixel is then determined according to the following criteria [2]:

- If E(x,y) $<$ T1, then the pixel is rejected and is not an edge pixel
- If E(x,y) $>$ T1 , then the pixel is accepted and is an edge pixel
- If T1$<$ E(x,y) $<$ T2, the pixel is rejected except where a path consisting of edge pixels connects it to an unconditional edge pixel with E(x,y) $>$ T2



(a) Threshold 0.1

(b) Threshold 0.3

(c) Threshold 0.5

(d) Threshold 0.7

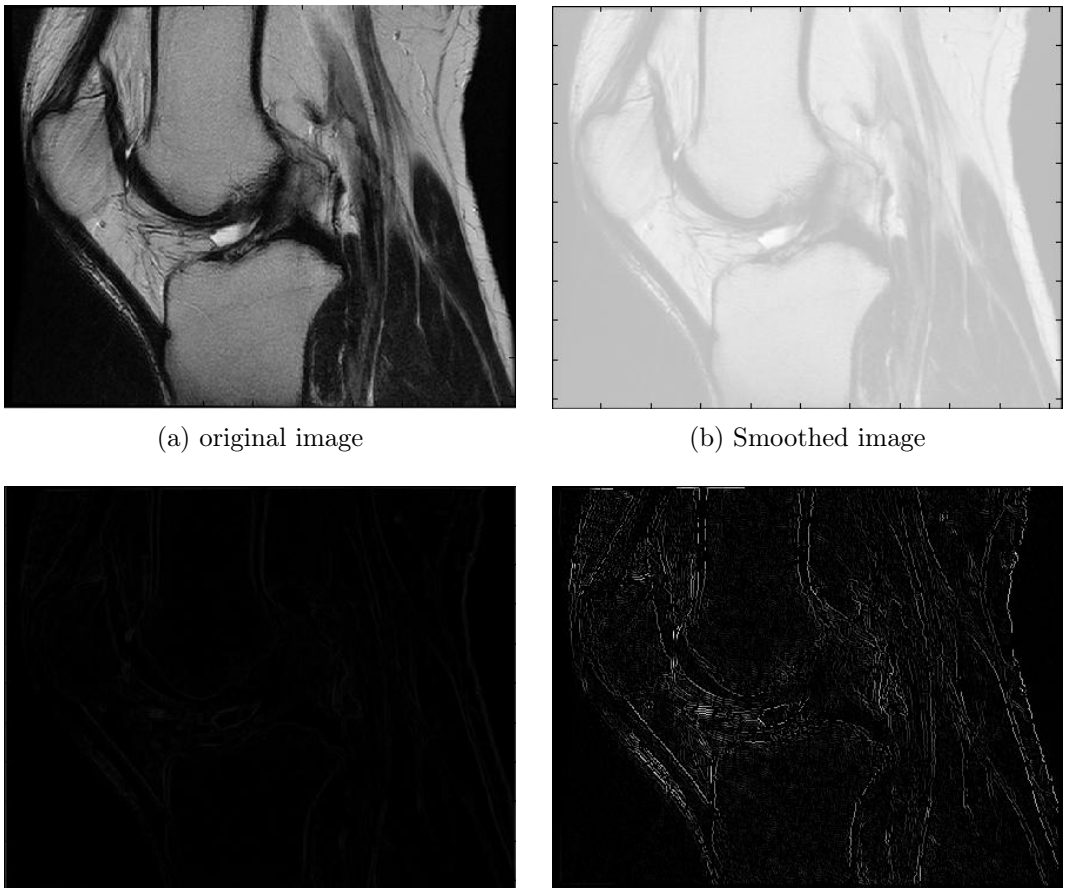
Fig. 3.11: Canny with different thresholds

4 SEGMENTATION MODULE

The segmentation module is called by the function edge and specifies the canny method. As mentioned before the canny method finds edges by looking for local maxima of the gradient of the image. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method is therefore less likely than the others to be fooled by noise, and more likely to detect true weak edges.

The function takes the selected slice as input and using a Gaussian filter modifies the input image by convolution with a Gaussian function. The use of the Gaussian filter is to have the overall effect of Gaussian blur. Gaussian blur or also known as Gaussian smoothing is the result of blurring an image by a Gaussian function and its main purpose is to reduce image noise and reduce detail. In our case it has the effect of reducing the images high frequency components. The intensity or the effect of the Gaussian blur on the selected image can be manipulated using sigma, the standard deviation of the Gaussian distribution. The size of the filter is chosen automatically, based on sigma.

After reducing the image noise using Gaussian smoothing, we take the smoothed image and find the strength of the edges by taking the gradient of the image. This is accomplished using Sobel masks in both directions. First we convolve the mage with the Sobel mask in the x direction and then we convolve the mage with the Sobel mask in the y direction. Those images are essential for finding the magnitude of the gradient of the image. After computing the absolute gradient magnitude at each point the function finds the direction of the edge and digitizes the edge direction so that it can be traced in an image.



(a) original image (b) Smoothed image

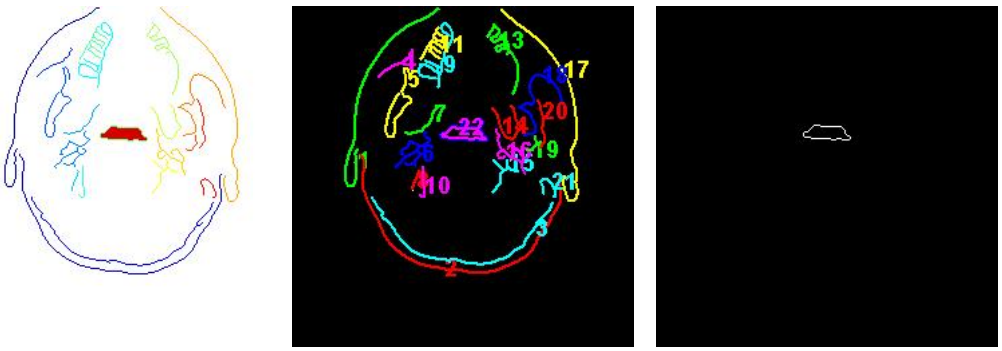
(c) Magnitude of the gradient (d) Edges

Fig. 4.1: Finding edges

Finaly the last task of the function is to track along the remaining pixels that have not been suppressed and threshold the image to identify the edge pixels. To accomplish that we use two distinct thresholds, a higher value T2 and a lower value T1. In our case the edge function gives you the ability to specify only the high value T2, the lower threshold value T1 is standart and kept unchanged. So if you specify a scalar for thresh, this scalar value is used for the high threshold and 0.4*threshold is used for the low threshold. The steps described above enable us to find the appropriate edges in the image simply by adjusting two variables, threshold and sigma. But that is not enough for the segmentation process which demands extracting from the image only the segments of interest. So by changing the values of threshold and sigma we can manipulate the image in a way that it is easier for us to move on to the execution of segmentation.

After we find the appropriate edges we can use the function bwlabel on the binary image which returns a matrix of the same size as the input image, containing labels for the connected objects in the binary image. After labeling the connected

objects we can visualize the individual edges in different color using the function label2rgb. Now that we are able to visualize them in different color we can select the edges that form segments in the image, isolate them and finally visualize them. The selection of the appropriate edges or segments is done by assigning each segment a different number by using the function bwboundaries. Functions bwbountries traces the exterior boundries of objects as well as boundaries of holes inside these objects, in the binary image. Now using the function label2rgb again we can visualize all enclosed segments or edges that form closed objects with different collor making the segmentation visually traceable. So by doing that and assigning each segment different number we can choose the number on an edge which encloses or is a part of a segment and extract it from the image. The final result of segmentation is again a binary image. As we can see from the image in which the edges are portrayed in different color, not all edges form enclosed segments. In fact most of the edges do not, making the whole process of segmentation more difficult. But the results of the segmentation will be discussed later.



(a) Edges and segments (b) Labeled edges (c) Final result

Fig. 4.2: Exraction of segments

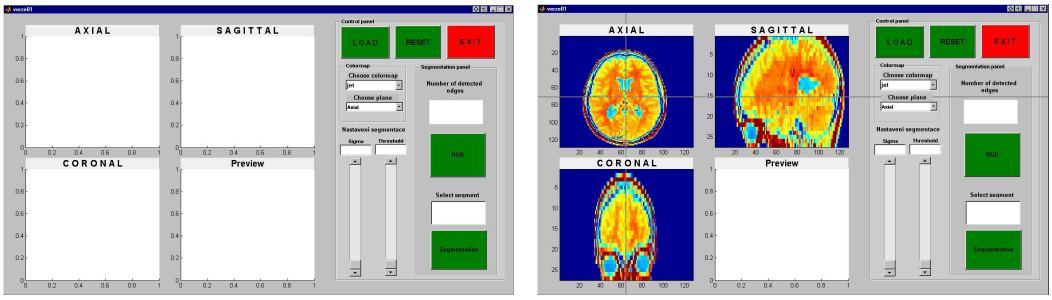
5 IMPLEMENTATION OF 3D VIEWER

In order to show the segmentation results and evaluate them we need to design and build 3D viewer, which would be able to retrieve the necessary data and display in all planes. In this chapter I will therefore deal with his creation and solution of its individual parts. Because of the inability of the canny method and generally of the edge detectors to segment directly a 3D object, it was necessary to construct the viewer in a way that we had to deal only with 2-diamensional images. The way we do that is that we load a volume of a 3D object and extract the information in a three plane so that we can implement the canny method more easily.

5.1 Loading data

The data that we display are simply loaded by clicking the load button and are stored in the database as an example to use in Matlab. They are stored in .mat format and are called using the function load. Data represent the entire volume of a head and are stored in a 4D matrix with dimensions 128x128x1x27. The first two dimensions are spatial. The third dimension is a dimension of color. The fourth dimension is also spatial. Regarding the direction of the cut, the first dimension represents the direction from the front to the back of the head so it represents the coronal plane. The second dimension indicates the direction from left to right, the sagittal plane. The fourth dimension indicates the direction from bottom to top of head, the axial plane.

Since the data are stored, as stated above in a 4D matrix it was necessary to reduce the matrix in 3D matrix for the correct display of the images. This is done using the function squeeze which ensures the reduction of dimensions by one. It was also necessary merely for aesthetic reasons to turn the sagittal and coronal plane by 90 °.



(a) 3D viewer (b) Loaded images

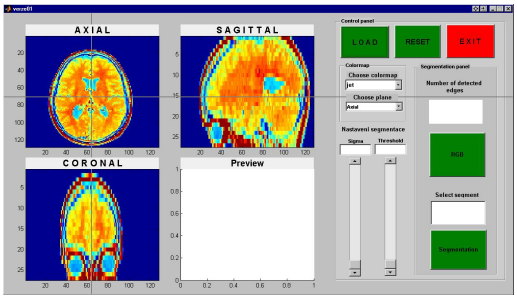
Fig. 5.1: Loading data

5.2 Browsing through the images

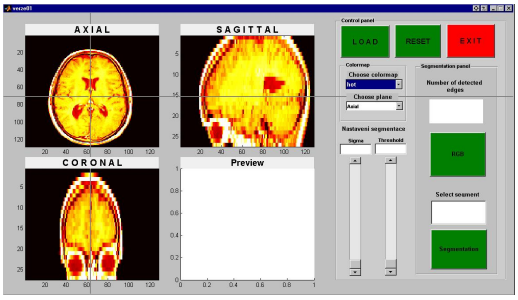
The ability to browse through the images is done by using function ginput witch returns the selected space coordinates. To be able to view another plane we need only to know one coordinate for example x, y, or z, which determines the area we want to portray. Using this method and the coordinates returned by the function ginput we are able to visualize the other two planes.

5.3 Graphical manipulation

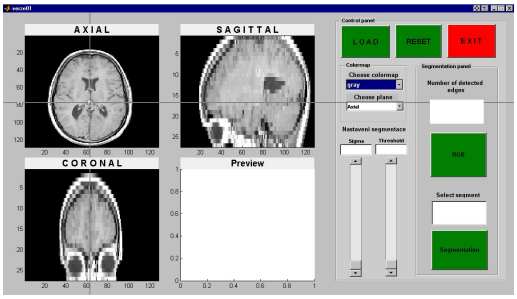
A colormap is a matrix of real numbers between 0.0 and 1.0. Each row is an RGB vector that defines one color. Viewer gives us the possibility to choose between four different colormaps, bone, gray, hot, jet. Bone is a grayscale colormap with a higher value for the blue component. This colormap is useful for adding electronic look to grayscale images. Gray returns a linear grayscale colormap. Hot varies smoothly from black through shades of red, orange, yellow, to white. Jet ranges from blue to red, and passes through the colors of cyan, yellow, and orange. The jet colormap is associated with an astrophysical fluid jet simulation from the National Center for Supercomputer Applications. The default colormap is set to jet.



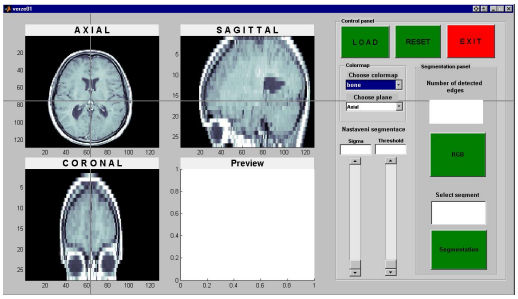
(a) Jet



(b) Hot



(c) Gray



(d) Bone

Fig. 5.2: Different colormaps

6 SEGMENTATION RESULTS

It would be very useful to be able to evaluate the performance of the edge detector since its role to the whole process of segmentation is critical. It could even be said that the entire segmentation of an image using edge detection is primarily based on the performance of its edge detector.

The evaluation of performance of edge detection in general is often made empirically simply by visual analysis of the detected edges. But in many cases that is not enough to judge the segmentation process based on the edge detectors, so I will also evaluate the performance of the edge detector Canny which is the basis of my segmentation process. The evaluation will be done quantitatively using phantoms which were affected with Gaussian white noise with different variance and phantoms added with salt and pepper noise with different noise density. This is done to underline the effect of added noise to the performance of the edge detector. Following the evaluation using phantoms I will evaluate the proposed edge detector empirically simply by visual analysis of the detected edges [11].

6.1 Evaluation on phantoms

This section focuses on the evaluation of the performance of the Canny edge detector algorithm using phantoms affected with different kinds and degrees of noise. Comparison of an edge map, obtained by a detector of edges, with its ground truth can be archived through a set of direct measurements, such as the number of edge pixels that were not classified as edge pixel, called the false negative (FN). The number of pixels, called false positive, pixels that were erroneously classified as edge pixels, called false pozitive (FP) and the number of correctly detected edge pixels, called true positive (TP). From these measures, the following statistical indices have been proposed: The percentage of pixels that were correctly detected(P_{co}) [11]:

$$P_{co} = \frac{TP}{max\,(N_I, N_B)}, \tag{6.1}$$

where N_I represents the number of edge points of the ideal image and NB the number of edge points detected.

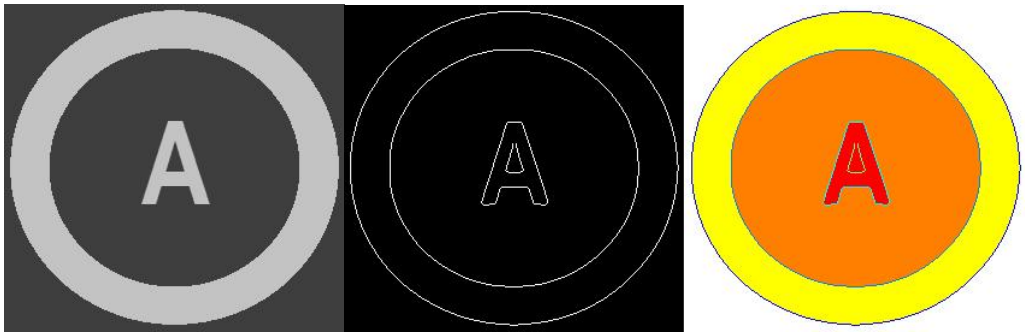
The percentage of pixels that were not detected (P_{nd}):

$$P_{nd} = \frac{FN}{max\,(N_I, N_B)}. \tag{6.2}$$

The percentage of pixels that were erroneously detected as edge pixels, ie the percentage of false alarm: (P_{fa}):

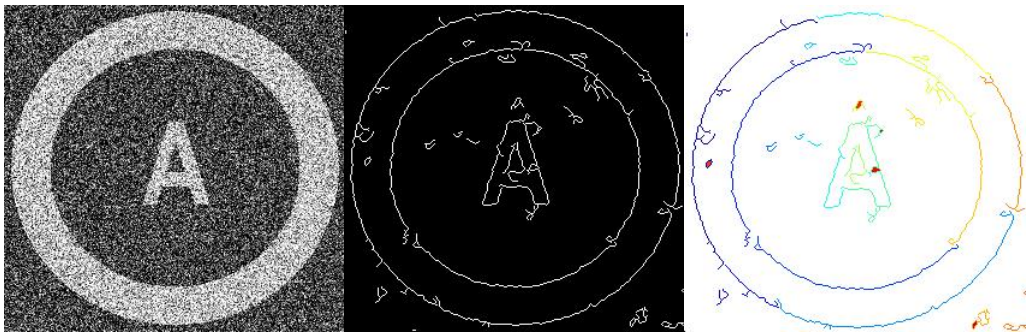
$$P_{fa} = \frac{FP}{max\,(N_I, N_B)}. \tag{6.3}$$

The following figures show the results of segmentation of phantom images affected with Gaussian white noise with increasing variance.



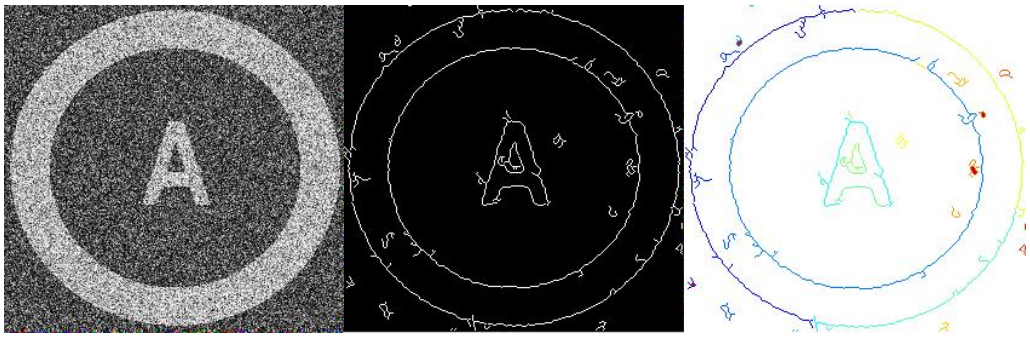
(a) (b) (c)

Fig. 6.1: Phantom without noise



(a) (b) (c)

Fig. 6.2: Phantom affected with Gaussian white noise, varianve 0.1



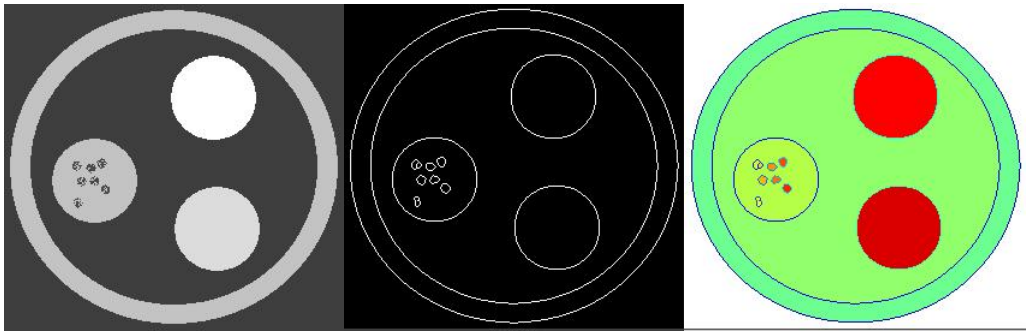
(a) (b) (c)

Fig. 6.3: Phantom affected with Gaussian white noise, varianve 0.2

Variance	P_{co} **[%]**	P_{nd} **[%]**	P_{fa} **[%]**
0	100	0	0
0.1	45.1	25.07	29.8
0.2	23.67	48.25	28.07

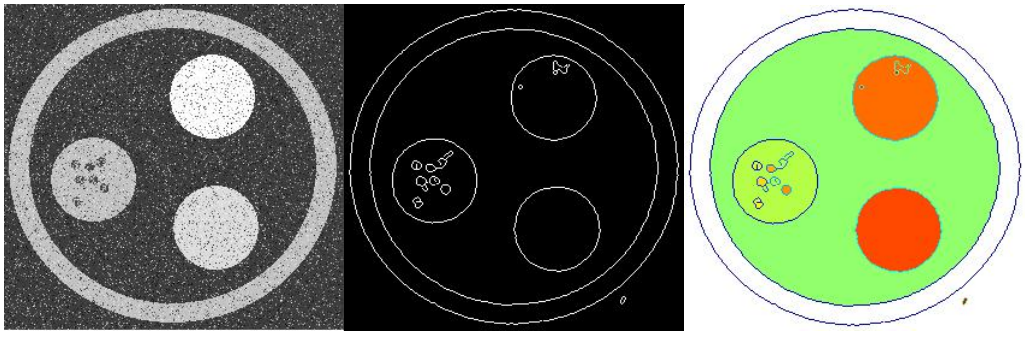
Tab. 6.1: Evaluation of the performance of Canny on images affected Gaussian white noise

The following figures show the results of segmentation of phantom images affected with salt and pepper noise to the image, with different noise densities.



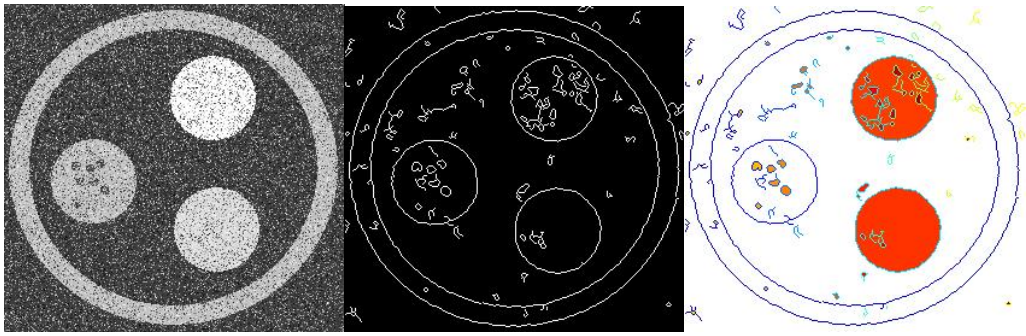
(a) (b) (c)

Fig. 6.4: Phantom without noise



(a) (b) (c)

Fig. 6.5: Phantom affected with salt and pepper noise, density 0.1



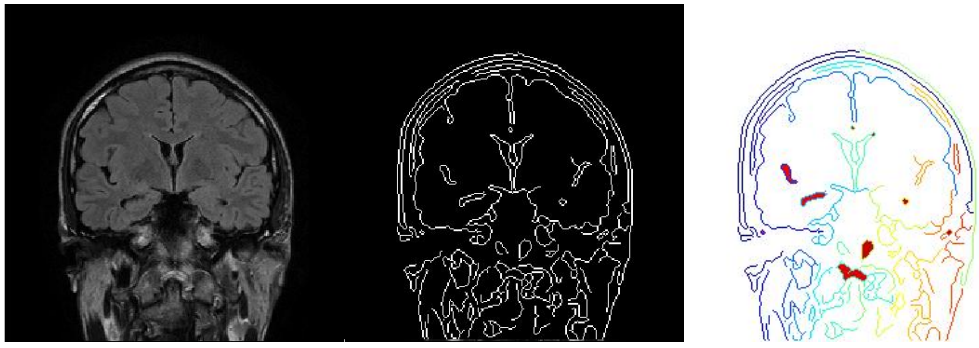
(a) (b) (c)

Fig. 6.6: Phantom affected with salt and pepper noise, density 0.2

Density	P_{co} [%]	P_{nd} [%]	P_{fa} [%]
0	100	0	0
0.1	28.6	66.88	4.5
0.2	19.28	43.74	36.97

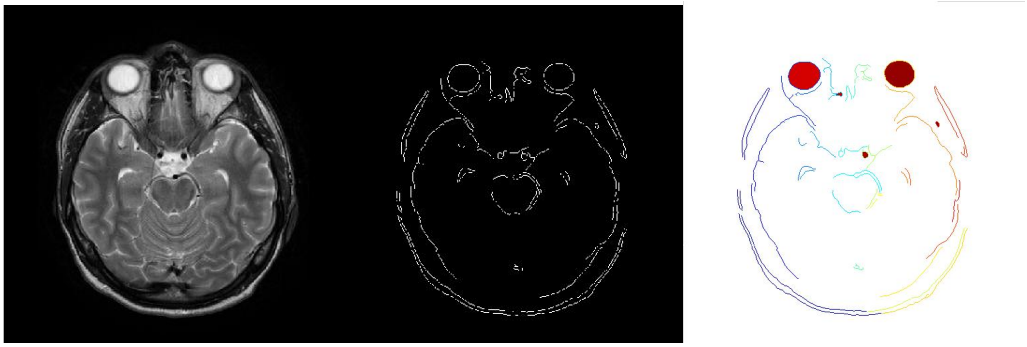
Tab. 6.2: Evaluation of the performance of Canny on images affected with salt and paper noise

6.2 Evaluation on clinical data



(a) (b) (c)

Fig. 6.7: Segmentation on clinical data



(a) (b) (c)

Fig. 6.8: Segmentation on clinical data

As we can see from the images above, it is rather difficult to find any meaningful segments that could be used for diagnostic purposes. Even if we have to deal with images with no artifacts and good signal to noise ration, the proposed edge detector is rather weak when used on clinical data and this due to the smooth change of pixel intensity values in the image. Canny edge detector is a first derivative edge detector which is based on the idea that in places where edge occurs, there is the greatest change in intensity, while in homogeneous areas the change and thus the first derivative s zero. And when an image is characterized by homogenous areas it is very difficult if not impossible to find meaningful enclosed edges that could be used in the process of segmentation.

Besides that, in cases of increased noise in the image or reduced signal to noise ration which is unavoidable when using clinical data, makes the challenge of detecting true edges a lot more difficult. So if we také into account the results of the evaluation of the performance of edge detectors, based on evaluation by means of quantitative indices, we will see that the measures defined in this section are in agreement with the qualitative analysis by visual inspection of what is commonly regarded as the best edges. Therefore, these measures may be useful and used to evaluate new more efficient algorithms for detecting edges.

7 CONCLUSION

In my thesis I present the basic segmentation techniques used in the field of medical image processing. The segmentation techniques were grouped into two main categories: boundary-based and region- based with my main concern being the boundary-based techniques and particularly the Canny algorithm witch I implemented. The proposed viewer is a 3D viewer build using matlab GUI and is able to load a volume of images representing the human head. From the whole volume which is in axial plane the viewer is able to extract the two remaining planes, sagittal and coronal. Through this viewer we are able using the segmentation module proposed in chapter 4 to segment the selected plane. The segmentation module was tested on phantom images affected with noise and its performance was evaluated quantitatively using the equations proposed in section 6.1. Looking at the results we can see that by increasing the variance of the white noise witch affects the image by 0.1, the percentage of pixels that are corectly detected as edges drops from 100% to 45.1%. Even worst results we take from using salt and papper noise, where by increasing the density of the noise by 0.1, the percentage of pixels that are corectly detected as edges, drops from 100% to 28.6%. Taking into account the evaluation visual and quantitative, we conclude that the segmentation method proposed in this thesis is unfit for successful segmentation of clinical MRI images. The main difficulty is to find any meaningful segments that could be used for diagnostic purposes. Even if we have to deal with images with no artifacts and good signal to noise ration, segmentation methods using as their basis different edge detectors will produce weak edges when used on clinical data and this is due to the smooth change of pixel intensity values in the image. Canny edge detector which is used in my segmentation module is a first derivative edge detector which is based on the idea that in places where edge occurs, there is the greatest change in intensity, while in homogeneous areas the change and thus the first derivative is zero. And when an image is characterized by homogenous areas it is very difficult if not impossible to find meaningful enclosed edges that could be used in the process of segmentation. We can see that there is a need for new better edge detection algorithms able to produce stronger finer edges that truly represent segments of anatomical structures in clinical data.

BIBLIOGRAPHY

[1] Information about DICOM files: available from URL: <ftp://medical.nema.org/medical/dicom/>

[2] DOUGHERTY, Geoff a Carolyn KAUT. *Digital image processing for medical applications. 1st ed. Cambridge: Cambridge University Press, 2009, 447 s.ISBN 978-052-1860-857.*

[3] CHRIS, Solomon a Toby BRECKON. *Fundamentals of Digital Image Processing: a practical approach with examples in matlab. 1. vyd. Oxford: Wiley-Blackwell, 2011, 328 s. ISBN 978-047-0844-731.*

[4] Catherine a Carolyn KAUT. *MRI in practice. 2nd ed. Malden, MA, USA: Blackwell Science, 1998, 326 s. ISBN 06-320-4205-2.*

[5] HAIDEKKER, Mark A. *Advanced biomedical image analysis. Hoboken, N.J.: John Wiley , c2011, 512 s. ISBN 04-706-2458-2.*

[6] SEIDL, Zdeněk a Manuela VANĚČKOVÁ. *Magnetická rezonance hlavy, mozku a páteře. 1. vyd. Praha: Grada, 2007, 319 s. ISBN 978-802-4711-065.*

[7] MAINI, Raman a Himanshu AGGARWAL. *Study and Comparison of Various Image Edge Detection Techniques. In: International journal of image processing. Malaysia: IJIP Journa, 2009. ISBN 1985-2304.*

[8] Willy, Wriggers. *Complex Numbers, Convolution, Fourier Transform. 2010.*

[9] HAIDEKKER, Mark A. *Advanced biomedical image analysis. Hoboken, N.J.: John Wiley , c2011, 512 s. ISBN 04-706-2458-2.*

[10] SUN, Xin a Xiaoxiao WANG. *Study of Edge Detection Algorithms for Lung CT Image on the basis of MATLAB. DOI: 10.1109/CCDC.2011.5968293*

[11] Adilson Gonzaga *Method to Evaluate the Performance of Edge Detector http://www.matmidia.mat.puc-rio.br/sibgrapi2009/media/posters/59906. pdfonline 2012-05-16*

LIST OF SYMBOLS, PHYSICAL CONSTANTS AND ABBREVIATIONS

DICOM Digital Imaging and Communications in Medicine

MRI Magnetic Rezonance Imaging

LoG Laplasian of Gaussian

NEMA NAtional Electrical Manufactures Association

GUI Graphics User Interface

P_{co} Percentage of pixels that were correctly detected

P_{nd} Percentage of pixels that were not detected

P_{fa} Percentage of pixels that were erroneously detected as edge pixels

TP Number of correctly detected edge pixels

FP Number of pixels erroneously classified as edge pixels

FN Amount of edge pixels that were not classified as edge pixel

LIST OF APPENDICES

A USER MANUAL

In this chapter I will deal with the various user factions that this program offers.

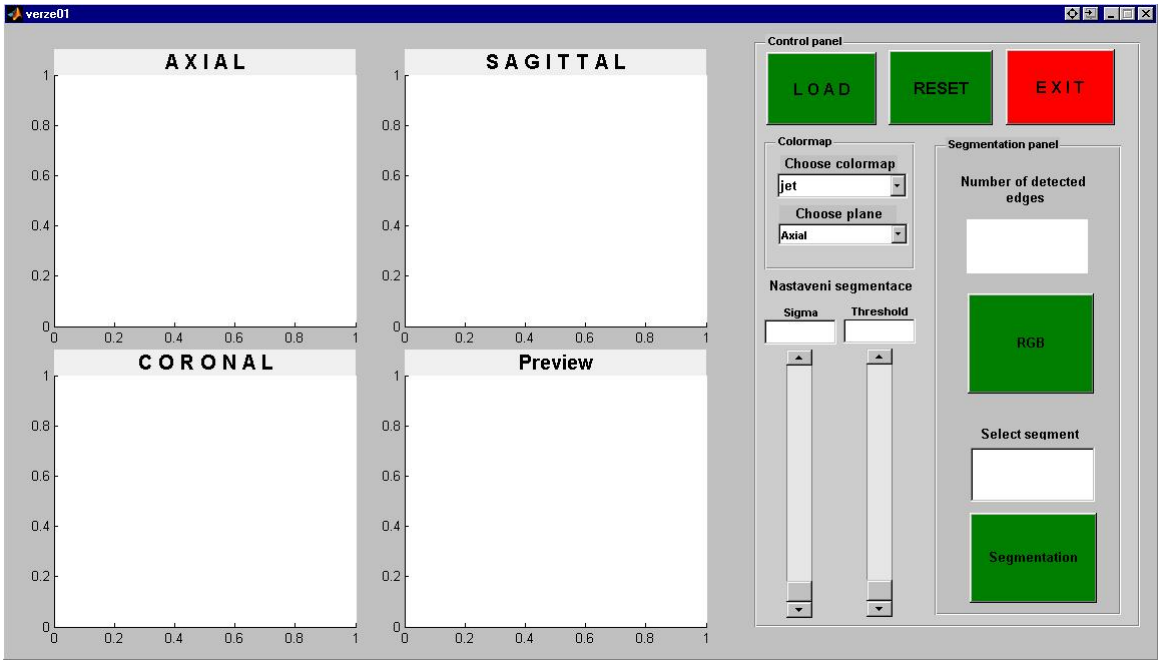


Fig. A.1: Main window

After starting the program, the main window appears. The working panel is placed to the right and bears all the elements essential when working with the data. Data are loaded using the LOAD button and the images are automatically loaded to the appropriate imaging fields. After loading the data we enter the browsing mode and the browsing cross is displayed. This mode enables us to freely browse through the images and choose areas that interest us. These areas are then shown in the remaining planes. After choosing the slices of interest we are ready to perform segmentation using the control panel. It is important to remember that after using the control panel you can return to the browsing mode only after using the reset button witch brings the program to its initial state.

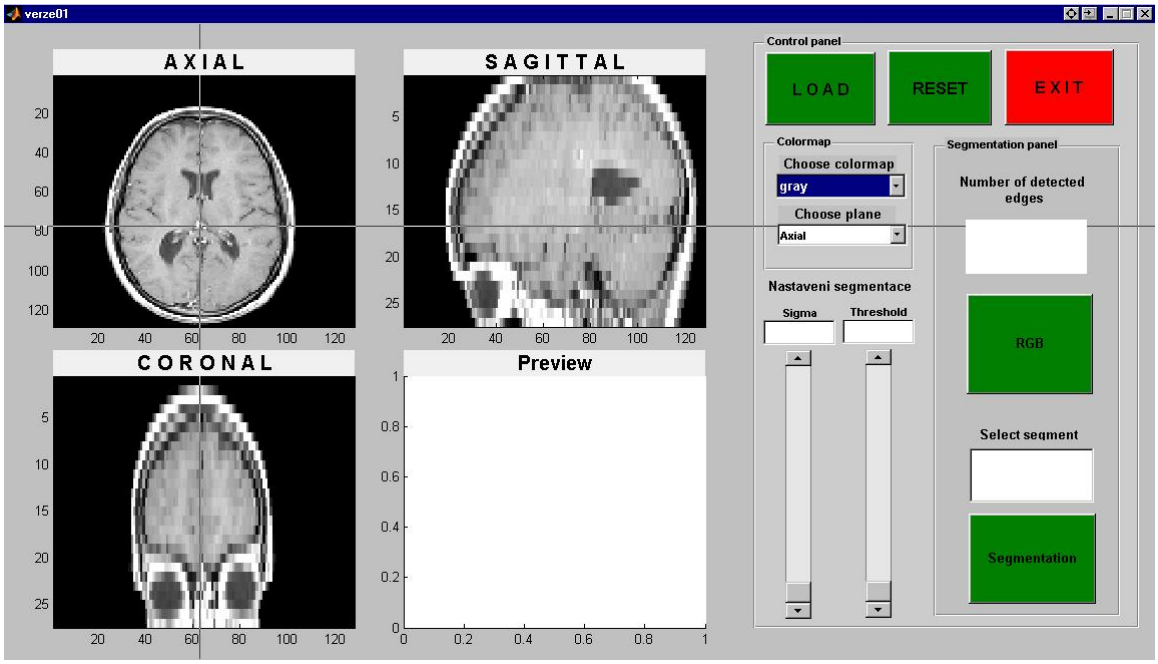


Fig. A.2: Browsing

Having selected the appropriate plane we can adjust our segmentation using the two sliders. The first slider adjusts the sigma factor of the Gaussian blur which affects the image. The second slider adjusts the big threshold value used by the Canny algorithm. The edges are displayed into the appropriate imaging field. Having adjusted the values of sigma and threshold the way we want by looking to the binary image we can now move on to the segmentation of the binary image by pressing the RGB button.

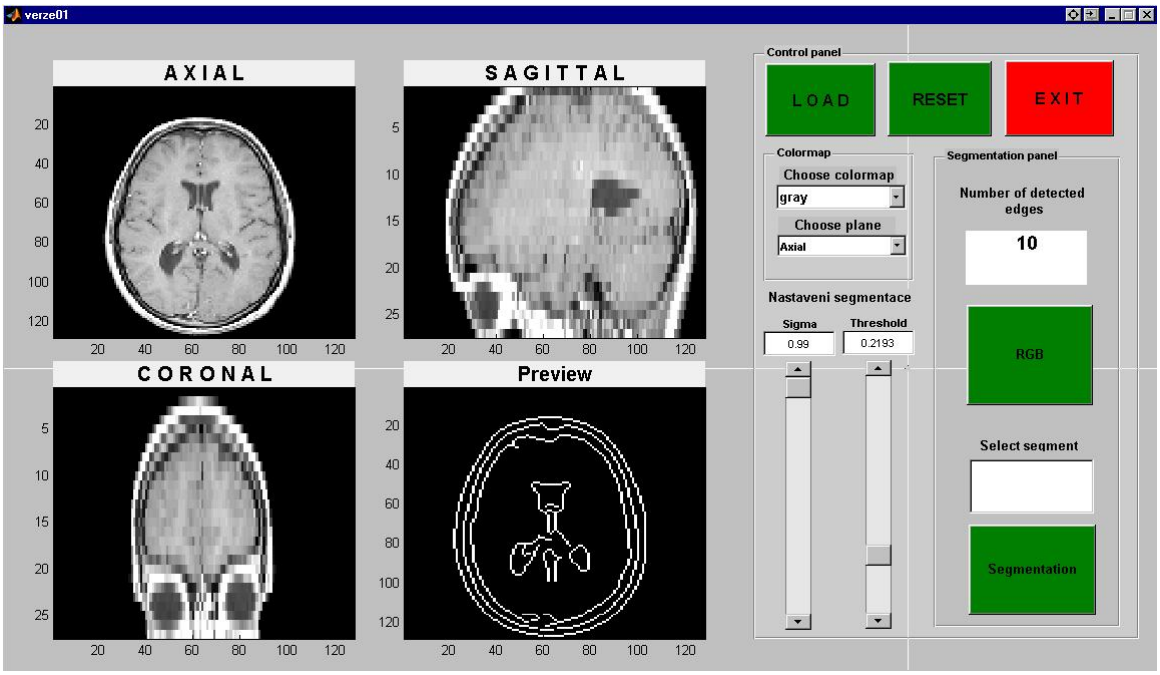


Fig. A.3: Segmentation preview

The RGB button creates another two imaging fields. One of the fields displays the edges found by the canny algorithm with different color each edge and fills the edges that represent enclosed segments. The other field displays each segment with a different number. Now having displayed each segment with different number we are able to choose the segment of interest by filling the appropriate blank with the number of the selected number. Doing that you are ready to push the Segmentation button which will extract from the image the selected segment.